

Patent
Attorney Docket: KUNZ/0001

IN THE CLAIMS:

Following is a replacement claim set.

1. (Currently Amended) A method for automatically generating flowcharting instructions from a source code comprising:

loading source code statements into statement records of a data structure, wherein each statement record contains only one of the source code statements;

identifying each branch statement contained in the statement records;

determining one or more branch destinations for each of the branch statements;

storing the one or more branch destinations in one or more destination records of the data structure, wherein the one or more destination records correspond to the statement record of the branch statement; and

identifying a statement type for each statement contained in the statement records;

exporting the data structure into a drawing program; and

displaying the flowchart.

2. (Original) The method of claim 1, further comprising:

maintaining a record management system of statement conventions comprising defined branch statement records and alternative branch statement records, wherein each alternative branch statement record is associated with one defined branch statement record.

3. (Original) The method of claim 2, wherein the record management system of statement conventions is selected from a database, a spreadsheet, an array, a set of coded rules in an applications program and combinations thereof.

4. (Original) The method of claim 2, further comprising:

finding any alternative branch statements in the statement records; and

replacing each of the found alternative branch statements with the associated defined statement.

Patent
Attorney Docket: KUNZ/0001

5. (Original) The method of claim 1, wherein the source code statements are written in a computer programming language selected from FORTRAN, COBOL, C/C++, PASCAL, Visual Basic, and PERL.
6. (Original) The method of claim 1, wherein the statement types are selected from decision, input, output, comment, computational, arithmetic, data, type, logic, start, loop, transfer and end.
7. (Original) The method of claim 1, wherein the statement types are selected from executable statements, non-executable statements, and comment statements.
8. (Original) The method of claim 1, wherein the statement types are selected from chain statements and branch statements.
9. (Original) The method of claim 1, further comprising:
maintaining a record management system of statement classifications comprising, for each source code statement type, a statement key, a statement format, destination locations, branch connector labels, a flowchart shape, and combinations thereof.
10. (Original) The method of claim 9, wherein the statement key comprises a word, a phrase, punctuation marks, other symbols, and combinations thereof.
11. (Original) The method of claim 9, wherein the record management system of statement classifications is selected from a database, a spreadsheet, an array, a set of coded rules in an applications program, and combinations thereof
12. (Original) The method of claim 1, wherein the step of identifying a statement type for each of the source code statements contained in the statement records further comprises:

Patent
Attorney Docket: KUNZ/0001

matching a statement key with each of the source code statements, wherein the statement key is associated with one statement type, and wherein the statement key comprises a word, a phrase, punctuation marks, other symbols, and combinations thereof; and

recording in the data structure the statement type for each statement in the statement records.

13. (Original) The method of claim 12, further comprising:

assigning a flowchart shape to each statement of the source code, wherein each flowchart shape is associated with the statement type of each statement of the source code; and

storing the assigned flowchart shape in a shape record in the data structure corresponding to the statement record.

14. (Original) The method of claim 1, wherein the data structure is selected from a spreadsheet, a database, a linked list, an array and combinations thereof.

15. (Original) The method of claim 1, wherein the data structure is created in a computer language selected from COBOL, C, PASCAL and C/C⁺⁺.

16. (Original) The method of claim 1, wherein the step of loading source code statements into statement records of a data structure further comprises:

assigning a unique statement number to each statement record; and

storing each unique statement number as a statement number record associated with each statement record in the data structure.

17. (Original) The method of claim 1, wherein the step of determining one or more branch destinations further comprises:

parsing each identified branch statement to find the one or more branch destinations designated within the branch statement;

Patent
Attorney Docket: KUNZ/0001

recording the one or more branch destinations into one or more branch destination records contained within the data structure, wherein the branch destination records are associated with the identified branch statement record; and

recording one or more branch labels into one or more branch label records contained within the data structure, wherein the branch label records are associated with the branch destination records.

18. (Currently Amended) The method of claim 1, ~~further comprising:~~

~~exporting the data structure into a drawing program, wherein the data structure contains statement records, statement number records, destination records, text branch label records, and shape records, the method further comprises: [[:]]~~

laying out shapes as defined in the shape records;
inserting text from the statement records into the shapes;
linking the shapes from the destination records; and
sizing the font to fit within the shapes; and
displaying the flowchart.

19. (Original) A method for generating a source code text file from a flowchart comprising:

assigning a shape name to each shape appearing on the flowchart, wherein the shape name is a statement number assigned to a first statement within the shape, and wherein each shape name is less than all subsequent shape names;

ending each source code statement within each of the shapes with a statement end character;

writing the statements contained within each shape to an ordered array with the associated shape names;

ordering the array from the least shape name to the highest shape name; and

writing each line of the ordered array to a text file.

20. (Currently Amended) A computer program product for generating flowcharting instructions from a source code comprising:

Patent
Attorney Docket: KUNZ/0001

loading instructions for loading source code statements into statement records of a data structure, wherein each statement record contains only one of the source code statements;

identifying instructions for identifying each branch statement contained in the statement records;

determining instructions for determining one or more branch destinations for each of the branch statements;

storing instructions for storing the one or more branch destinations in one or more destination records of the data structure, wherein the one or more destination records correspond to the statement record of the branch statement; and

identifying instructions for identifying a statement type for each statement contained in the statement records;

exporting instructions for exporting the data structure into a drawing program; and
displaying instructions for displaying the flowchart.